Collaboration and the Knowledge Economy: Issues, Applications, Case Studies
Paul Cunningham and Miriam Cunningham (Eds)
IOS Press, 2008 Amsterdam
ISBN 978–1–58603–924-0

Design and Reference Implementation of a Distributed URN Registry: URNreg

Victoriano GIRALT[1], Diego LOPEZ[2], Cándido RODRIGUEZ[3], Milan SOVA[4]
[1]*University of Malaga, Blvd. Louis Pasterur, 33 Ed. SCAI Campus de Teatinos, Malaga, 29071, Spain*
Tel: +34 95 2132366, Email: victoriano@uma.es
[2]*Red.ES/RedIRIS, Avda. Reina Mercedes s/n Ed. CICA, Sevilla, 41012, Spain*
Tel: +34 95 5056621, Email: diego.lopez@rediris.es
[3]*Red.ES/RedIRIS, Avda. Reina Mercedes s/n Ed. CICA, Sevilla, 41012, Spain*
Tel: +34 95 5056613, Email: candido.rodriguez@rediris.es
[4]*CESNET, Zikova 4, Praha, 160 00, Czech Republic*
Fax: +420 2 2432 0269, Email: sova@cesnet.cz

Abstract: Resource identifiers are critical in our technical environment nowadays. Uniform Resource Names (URNs) are intended to serve as persistent, location-independent resource identifiers and are designed to make it easy to map other name-spaces (that share the properties of URNs) into URN-space. Current instances of URN registries are usually implemented as a set of web pages serving simple enumeration of registered URNs and sometimes their simple descriptions. This paper describes the design and the implementation of URNreg, a novel service which provides a way to get information of a registered URN to both human and non-human users. Also, it supports a distributed delegation model so it is able to get the registry information for delegated namespaces.

1. Introduction

Identity federations are distributed in themselves. Apart from those services enabling the discovery of where to assess user identities, which are not essential and need not be unique, they do not require any central point or authority.

On the other hand, many identity federations, especially in the academic environment, have resorted to URN based values for attributes when it becomes necessary to carry contextual information alongside the value itself.

The use of this type of values is exploding. However, interoperability requires the meaning of the values, at least, to be understood by all the federation participants. Even, in some cases, to be the same for a given attribute, be it at the federation level or some other levels, whether geographical or organizational.

This paper presents the policy and technical mechanism that have been designed to achieve both a distributed control of URNs and the capacity to use such values at an operational level.

The software has been put into production in some projects over the European Research and Academic Network.

2. Basic concepts

Uniform Resource Names (URNs) [1] are intended to serve as persistent, location-independent, resource identifiers. The URN syntax provides a means to encode character data in a form that can be sent in existing protocols, transcribed on most keyboards, etc.

URNs are specially formatted strings with three basic parts:

- The urn*:* prefix
- The namespace identifier or NID
- The namespace specific string or NSS.

The NSS and the NID are separated by a colon. NSS may be further subdivided into sub-namespaces, possibly managed by different authorities, leading to a hierarchical structure.

Delegations can happen at any point of the hierarchy, and each one should have a corresponding point of publication of assigned values and subsequent delegations. This has been usually done by means of hand maintained web pages, without any service for getting automatic resolution of URNs.

Internet Assigned Numbers Authority (IANA) is the central repository for URN registries [2] containing the highest level of registered and delegated URNs.

3. Service design

The solution proposed in this paper provides a service that can be used for finding registries and it has been designed around registered and delegated URNs.

The protocol of URNreg is based on REST web services [3] with the requirement to provide capabilities for interacting with both human and non-human users. HTML pages are showed to the former group of users whereas XML messages are served to the latter.

The protocol uses basically HTTP requests using GET or POST methods to the URL of a particular URNreg web service. The parameters are being passed the usual way, i. e. as parts of the query string (for the GET method) or within the request body (for POST). Each web service specifies in its definition which parameters it expects to receive and their valid acceptable values.

The defined web services in the specification of the protocol of URNreg are:

- List URNs: request to list URN descriptions or delegation records stored at the particular URNreg server. The service recognizes the following optional parameters:
  - Format: specifies whether the list should be returned in human- or machine-readable format.

  Acceptable values:
    - Human (default) for human readable response (usually HTML)
    - Machine form an XML message
  - Type: specifies which type of records should be listed

  Acceptable values:
    - Description (default) for requesting the description of registered URNs
    - Delegation for requesting the delegation records
- Search for a URN: request for the information about a URN. In case of the namespace of that urn is delegated, it queries the remote registry and returns the obtained information. This service accepts the following parameters:
  - Format: specifies the requested format of the response. The parameter is optional

  Acceptable values:
    - Human (default) for human readable response (usually HTML)
    - Machine form an XML message
  - urn: the requested urn.

Figure 1 describes the scenario after deploying URNreg. An application, with human or non-human interactions, sends a query to the list or search web services. But, in case asks for the information of an URN delegated to another URNreg instance, the service will get that information on behalf of the application. The URNreg front-end is able to perform searches on delegated namespaces. Please note, that the List and Search services behave differently concerning the delegation. While the former just lists the information about

delegations the latter actually returns the description even for URNs delegated to other authorities.

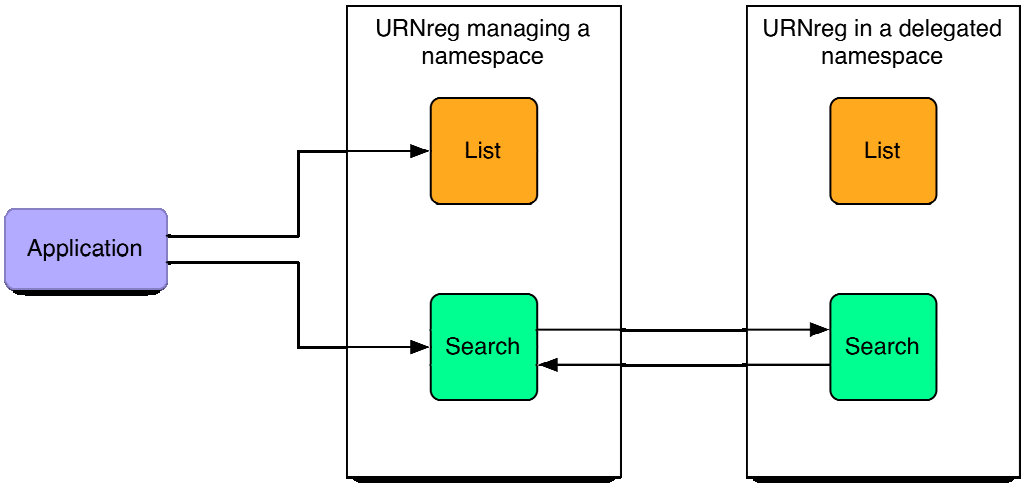

Figure 1: the scenario of URNreg

Each REST request is answered by a REST response, which in case of a machine-readable format is an XML message in a format specific to the particular service. Table 1 provides an example response to the request

http://webserver/urnreg/search/?urn=urn%3Ageant%3Aeduroam&format=machine

```
<?xml version=’1.0’ encoding=’UTF-8’ ?>
<Response>
  <urn value="urn:geant:eduroam">
    <descriptions>
      <description value="This is the root URN for eduroam components"/>
    </descriptions>
    <presentations>
      <presentation value="URN raíz para el espacio de nombres de eduroam"
lang="es"/>
    </presentations>
  </urn>
</Response>
```

Table 1: an example of response of URNreg

The data stored when an organization registers an URN is usually quite basic so far, saving only the value of the URN, when it has been registered and who has requested it. URNreg proposes to store more information about the registration of an URN in order to get a full URN resolution service.

The information issued for a registry and for a delegation is different. The registration record consists of the following information:

- URN value: the registered URN value.
- Description: description of the URN value.
- Presentation: human readable representation for the URN value. This attribute allows for multilingual values and its main goal is users can get a well explanation of what the URN is representing.
- Delegated namespace: in case of URNreg has resolved the delegation of the URN and get its data from the delegated namespace service, it says which web service has been used for.

- Contact person: who has requested the registration of the URN.

On the other hand, the information issued for a delegated URN has the following elements:
- URN value: the delegated URN value.
- Owner: the Internet domain of the institution that owns the URN namespace.
- Register web application: the URL of the web application, or the entry link for a set of HTML pages, used for registering URNs in that delegated namespace.
- Policy document: the URL of the document expressing the policy of that delegated namespace.
- Web services for accessing its URNs: the URL base for the URN web services.
- LDAP URI for accessing its URNs: an optional URI which specified how to get access to an LDAP where URNs are stored.
- Contact person: who has requested the registration of the URN.

4. Reference implementation

A reference implementation has been developed in PHP using the siLeDAP framework [4] for accessing LDAP directories through web services. LDAP has been selected as object store back-end for its performance in search and read operations and for the ease of definition of object classes and attributes. A new schema for storing registered and delegated URNs has been published.

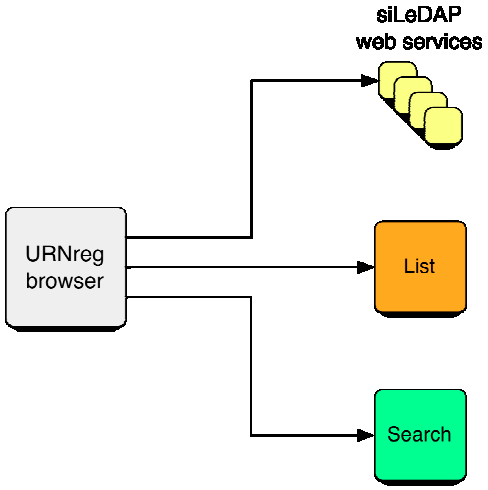

Figure 2: the architecture of the URNreg browser

Also, a PHP user interface has been developed that communicates with the web services to offer a human interface for querying and registering the URNs. This interface allows a user to query all registries from one point following the pointers for machine interface in the delegation objects. The user is also presented with the machine interface links to the web interface of the delegated registries responsible for the namespace. As shown in figure 2, this user interface gets the list of registered and delegated URNs by interacting the web services defined by URNreg.

Using the siLeDAP API, we have developed another set of web services in order to manage the registration of new URNs. Using them, it is possible to register an URN or request a namespace delegation using the URNreg browser.

The human interface is, of course, a web application, that queries the registry that manages the particular namespace. Once a namespace delegation entry is reached, there are two possible paths to follow:
- In order not to confuse the user, the application queries the remote registry and presents the data to the user with the local look and feel. This is probably a less confusing experience for the user, that even allows to present the user with a localized version of

the value, if it is available.

- The application presents the URL of the new registry human interface to the user. Following such link the user will arrive to the registry entry page. This has the risk of landing the user on a page she does not understand, but clearly presents the identity of the namespace owner. This way of operation should follow the standard i18n procedures for selecting the user language and, if it is unavailable either on the request or inside the registry, it must present information in English.

5. Usage

The SChema Harmonisation Committee (SCHAC) schemas [5] mandate the use of URNs as values for certain attributes, allowing for hierarchically assigned encodings, that decouple national- or community-wide variations of otherwise similar values. A clear example of this is schacHomeOrganizationType that describes the sort of institution a certain person is linked to. The simple value "university" has slight but significant differences in meaning in different countries. A URN encoding allow for a consistent spaces of values across national borders and leave each local system the freedom to assign specific, well-defined mapping rules.

The Bologna Process aims to create a European Higher Education Area by 2010, in which students can choose from a wide and transparent range of high quality courses and benefit from smooth recognition procedures. The internal administrative actions made by European universities will use the URNs registered by SCHAC initiative, which is provided by the URNreg of TERENA, since those organizations must work with same definitions around whole Europe.

The root of the namespace for SCHAC attribute values is operated by TERENA, and from the above example it is easy to see that it requires a comprehensive system to manage both common and specific values, including delegations. URNreg combines the ease of user of its Web interface for human administrators, with the possibility of using the Web services from applications using the SCHAC attributes in order to establish correctness and appropriate mappings of their values. The delegation protocol permits applications to query a single point that will appropriately redirect them to the URNreg instance able to satisfy the request.

The URNreg distributed registry is used in the eduGAIN interfederation infrastructure [6] inside the GÉANT2 project. This project uses URNs for providing unique, persistent and well-managed identifiers to the components, either resources or identity repositories, within the participating federations. These identifiers, along with data about when, how, what to and whom for they were assigned, are managed by the eduGAIN Naming Registry.

The values are used in a distributed environment, applying them in all the process required for establishing trust inside the eduGAIN infrastructure. For example, since these URNs serve as identifiers for eduGAIN entities, they are included in their X.509 certificates, as an extension mandated by the eduGAIN certificate profiles. The URN registry provides an invaluable resource for the Certificate Authorities to validate the Certificate Signing Requests, while allowing each participant federation to retain control on the naming schema applied to the components operated inside it. Verifying the meaning, eligibility, and correctness of an attribute value is just one click away.

6. Related work

Both the authors and the working groups on federations in which they participate had no knowledge of any similar work, so we had to develop a totally original system.

Some URN resolution services have developed so far, such as DiVA [7] project, which uses URNs to map electronic resources to URLs and as a primary key of the publications

stored in DiVA. But these kinds of service do not provide a generic service.

Also DDDS, an Internet protocol for finding distributed resources using the resource name itself and DNS is being considered for using in URNreg in future releases.

7. Conclusion and future work

The system described in this paper has proved useful for production federations to control the URN explosion and to ease the dissemination of values and their meanings. The system will be deployed during 2008 in many European NREN operators to administer their delegated namespaces from prefixes belonging to projects such as the already mentioned eduGAIN and SCHAC. The Australian Access Federation has also shown interest in the system to manage their URN namespaces.

References

[1] R. Moats, "URN syntax," RFC 2141, IETF, 1997.
[2] "The official IANA registry of URN namespaces," http://www.iana.org/assignments/urn-namespaces.
[3] R. Fielding, "Architectural Styles and the Design of Network-based software Architectures," http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.
[4] "The siLeDAP project", https://forja.rediris.es/projects/siledap/
[5] "TF-EMC2 SCHAC project," http://www.terena.nl/activities/tf-emc2/schac.html.
[6] "Deliverable DJ5.2.2,2: GÉANT2 AAI Architecture and Design," GN2-JRA5 deliverable, January 2007.
[7] E. Müller, U. Klosa, P. Hansson, S. Andersson, E. Siira, "Using XML for Long-term Preservation. Experiences from the DiVA Project," Proceedings of the ETD 2003: Next Steps - Electronic Theses and Dissertations Worldwide. The Sixth International Symposium On Electronic Theses and Dissertations, the Humboldt-University in Berlin, Germany, 21 - 24 May 2003.
[8] M. Mealing: Dynamic Delegation Discovery System (DDDS) Part One: The Comprehensive DDDS, RFC 3401, IETF, 2002.